Discussion Results of 14-04 crossValidationCVFiX

Layer (type) Output Shape Param # Connected to

==================================================================================================

input\_3 (InputLayer) (10, 10, 2400) 0

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

attention\_vec (Dense) (10, 10, 1) 2401 input\_3[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dot\_3 (Dot) (10, 1, 2400) 0 attention\_vec[0][0]

input\_3[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lambda\_5 (Lambda) (1, 10, 2400) 0 dot\_3[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_5 (Dropout) (1, 10, 2400) 0 lambda\_5[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_3 (LSTM) (1, 10, 64) 631040 dropout\_5[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dropout\_6 (Dropout) (1, 10, 64) 0 lstm\_3[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lambda\_6 (Lambda) (10, 64) 0 dropout\_6[0][0]

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_3 (Dense) (10, 1) 65 lambda\_6[0][0]

==================================================================================================

Total params: 633,506

Trainable params: 633,506

To think about   
To do  
Working on

No Regularization No Dropout

* As we can see we have ‘Problem with find accuracy’ always with very high and only with small memory using very low learning rate. In case of very low learning rate the problem should be that the validation loss is always decreasing (small learning so increase it or increase the # of epochs). In case of very high learning rate the problem is due to ‘too long jump’ the optimizer do caused by the strong learning rate (momentum or decay could help in this situation)
* Best results are achieved with very small learning rate and high memory. I’m not sure why a such small learning rate could work (i’m using adam optimizer)
* I’ve tried also with bigger memory like 756,1024 but the results seems not increasing a lot
* 64 best results learning =10-6 5970 against 57
* 128 best results learning=10-6 5975 against 57
* 256 best results learning=10-6 56 against 57
* Augmenting the memory the ‘Problem in find max acc’ at high learning rate decreases why?
* In best results often i have ‘Problem in find max acc’ probably increasing number of epochs, i should try to plot it to see the learning curves in each fold

Regulatization No Dropout

* In case of regularization trend results seems the same but with worse results always under the baseline. Is it caused by model too simple? Let’s try with more complicated one (increase memory) working on it.. Worked on int but simulation stopped before doing model selection also with dropout>0 for server’s problem
* Also in this case seems 10-5 or 10-6 the best learning rate